DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CALIFORNIA 93943 8002

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Mathematical Model of a Marine Corps Amphibious
Landing Based upon Intelligence Estimates

by

Catherine Ann Johnson

September 1991

Thesis Advisor: William Walsh

Approved for public release; distribution is unlimited.

T257822

Unclassified

Security Classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification UNCLASSIFIED	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report Approved for public release; distribution is unlimited.
2b Declassification/Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School 6b Office Symbol *(If Applicable)* OR	7a Name of Monitoring Organization Naval Postgraduate School
6c Address *(city, state, and ZIP code)* Monterey, CA 93943-5000	7b Address *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a Name of Funding/Sponsoring Organization 8b Office Symbol *(If Applicable)*	9 Procurement Instrument Identification Number
8c Address *(city, state, and ZIP code)*	10 Source of Funding Numbers

Program Element Number	Project No	Task No	Work Unit Accession No

11 Title *(Include Security Classification)* MATHEMATICAL MODEL OF A MARINE CORPS AMPHIBIOUS LANDING BASED UPON INTELLIGENCE ESTIMATES

12 Personal Author(s) Catherine Ann Johnson

13a Type of Report Master's Thesis	13b Time Covered From To	14 Date of Report *(year, month,day)* 1991, September	15 Page Count 58

16 Supplementary Notation The views expressed in this paper are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes

Field	Group	Subgroup

18 Subject Terms *(continue on reverse if necessary and identify by block number)*
Transshipment network; amphibious landing; Intelligence Preparation of the Battlefield (IPB); GAMS

19 Abstract *(continue on reverse if necessary and identify by block number*

This thesis develops a model to assist in the intelligence gathering and operational analysis of an amphibious landing assault. It utilizes major intelligence considerations of the Intelligence Preparation of the Battlefield (IPB) process to aid the force commander in decisions required prior to the assault. The algorithm for the model is written in the FORTRAN programming language. Input into the model involves the weather, terrain, sea state, and resistance the force can expect to encounter during each phase of the assault, along with the troop requirement to meet the objective. The FORTRAN program uses the input data to produce a transshipment network which will be optimized and solved by the General Algebraic Modeling System (GAMS). Output from GAMS is the number of Marines to be assigned to each assault objective. A typical amphibious landing network is set up in the thesis and output is analyzed in an effort to demonstrate the usefulness of this decision-making tool.

20 Distribution/Availability of Abstract ☒ unclassified/unlimited □ same as report □ DTIC users	21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual W. Walsh	22b Telephone *(Include Area code)* (408) 646-3113	22c Office Symbol OR/Wa

Approved for public release; distribution is unlimited

Mathematical Model of a Marine Corps Landing based upon Intelligence Estimates

by

Catherine Ann Johnson
Captain, United States Marine Corps
B.S., University of Southern California, 1984

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

From the

NAVAL POSTGRADUATE SCHOOL
September 1991

ABSTRACT

This thesis develops a model to assist in the intelligence gathering and operational analysis of an amphibious landing assault. It utilizes major intelligence considerations of the Intelligence Preparation of the Battlefield (IPB) process to aid the force commander in decisions required prior to the assault. The algorithm for the model is written in the FORTRAN programming language. Input into the model involves the weather, terrain, sea state, and resistance the force can expect to encounter during each phase of the assault, along with the troop requirement to meet the objective. The FORTRAN program uses the input data to produce a transshipment network which will be optimized and solved by the General Algebraic Modeling System (GAMS). Output from GAMS is the number of Marines to be assigned to each assault objective. A typical amphibious landing network is set up in the thesis and output is analyzed in an effort to demonstrate the usefulness of this decision-making tool.

Thesis
T5933
c.1

TABLE OF CONTENTS

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

I. INTRODUCTION

The purpose of this thesis is to develop a model that will assist the Marine Corps intelligence officer and force commander in the decision analysis of the troop movement required for an amphibious assault operation. Many times intelligence information has been misdirected, thus not making it available to those units which require it for a successful combat campaign. This difficulty can be attributed to two possible theories. The first is "information overload," where the incoming intelligence information is too excessive for an intelligence unit to properly analyze and effectively distinguish important intelligence from the unimportant. A second theory and one which has also been a difficulty in the Marine Corps is the separation between intelligence and operational units. Efforts are ongoing to bridge this gap between these entities. There have been attempts to integrate intelligence units such as the Surveillance, Reconnaissance and Intelligence group (SRI) into the operational staff. Also the Marine Corps formed an "All-Source Fusion Center" which was designed to analyze information and produce the necessary intelligence.

This thesis is an analytical effort to unite the intelligence and operational units and provide the force commander with alternative means to plan an assault based upon the intelligence estimates. It begins by defining the problem and describing the intelligence procedures the Marine Corps currently utilizes when preparing for combat. This will be followed by a description and implementation of the intelligence model. The conclusion contains an analysis of the data and suggestions for future enhancements.

A. PROBLEM STATEMENT

The Marine Corps has recently assumed a new approach to warfighting called the Marine air-ground task force, or MAGTF, at both the operational and tactical levels. The MAGTF doctrine emphasizes the employment of all elements of the force under a single commander, thereby obtaining unity of effort. Its organization by task enables the commander to tailor the force to a specific contingency. The success of this integrated air-ground team depends upon identifying and exploiting critical enemy weaknesses. Intelligence thus assumes a crucial role in the identification of enemy capabilities. Observation and orientation of the enemy must occur prior to any battle planning and execution.

Many factors influence the pre-battle assessment, and intelligence preparation of the battlefield. One significant complaint during this phase is the continuous misuse and often neglect of intelligence units (IU). The tasks assigned an IU are often too voluminous and multifaceted to accomplish accurately in a timely manner. The IU has also been slow to adopt MAGTF warfare and seems to be more concerned with assimilating new technology than with integrating a new warfighting concept. Efforts have been made to resolve the problems but some feel there still exists a disconnect between the intelligence and operations community [Ref. 1].

For MAGTF warfare to achieve its goal, operations and intelligence must be an inseparable, cohesive unit. Intelligence should be guiding operations and, in the same vein, operations should be exploiting intelligence. The intelligence officer must be privy to the scheme of maneuver and the commander's intentions must be shared. When this occurs the intelligence community will have the insight it needs to drive the intelligence effort in the right direction. Operations will benefit from the crucial intelligence it needs to successfully implement a battle plan. [Ref. 1]

This thesis develops a planning model that will aid the intelligence efforts and provide a communications link between the intelligence and operations communities. In order to use this model, information concerning scheme of maneuver as well as intelligence estimates of the battlefield will be needed. The purpose of this model is to help bridge a gap and alleviate a communication difficulty which exists between Marine intelligence and operational units.

B. MARINE CORPS INTELLIGENCE AND THE IPB PROCESS

Intelligence has been vital to combat operations as early as 500 B.C. when Sun Tzu stated "Know the enemy, know yourself; your victory will never be endangered. Know the ground, know the weather; your victory will then be total." [Ref. 2]

Marine Corps history has proven Sun Tzu's wisdom many times from the island campaigns of WWII to the recent engagement with Iraq. **Recon-pull**, a term describing intelligence efforts and the search and exploitation of the enemy weakness, is at the heart of the MAGTF mission. There are two phases of Recon-pull or simply reconnaissance: reconnaissance prior to battle involving long-range intelligence gathering, and battle reconnaissance which is "reconnaissance by fighting." The model developed is primarily concerned with the long-range reconnaissance and intelligence gathering. [Ref. 3]

The Marine Corps has developed a systematic and continuous approach to reconnaissance prior to battle. It is known as **Intelligence Preparation of the Battlefield** (IPB). IPB provides an analysis of the enemy, weather and terrain in a specific location. The weather and terrain information is integrated with enemy doctrine to determine possible courses of action and enemy vulnerabilities. A graphical representation of the IPB process is shown in Figure 1. The following is a description of the five functions in the IPB process. [Ref. 4]

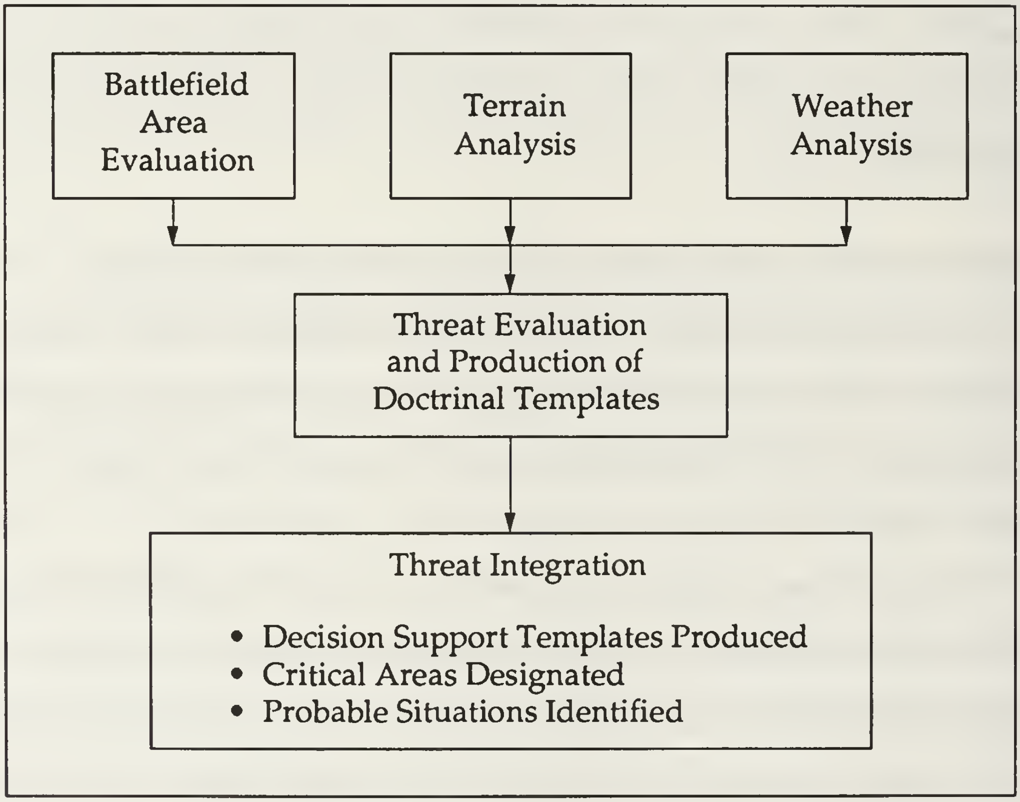
Battlefield Area Evaluation
Terrain Analysis
Weather Analysis
Threat Evaluation and Production of Doctrinal Templates
Threat Integration
• Decision Support Templates Produced
• Critical Areas Designated
• Probable Situations Identified

Figure 1. Intelligence Preparation of the Battlefield (IPB)

- Battlefield Area Evaluation (first function). This function determines the boundaries and areas of responsibilities for each commander. The **area of interest** is the entire battle area based on the estimate of the situation and must extend in all directions to safeguard the command from surprise. The area **of operations** represents an area defined by a boundary or geographical feature in which authority has been assigned to a specific commander. The **area of influence**, which is inside the area of operations, is the space where a commander is capable of engaging the enemy. In all of these areas the commander must view the space in four dimensions: width, depth, airspace, and time.

- Terrain Analysis (second function). This function analyzes the military aspects of terrain and the effects it will have on friendly and enemy abilities to move, engage and communicate. Movement of troops is generally regulated by obstacles and avenues of approach. Key terrain and fields of fire will stipulate when and where the bases of fire are placed and the strategic communication points. To assist the intelligence officer in the terrain analysis, a data base called the **Tactical Terrain Analysis Data Base** (TTADB) was established by the Defense Mapping Agency. The data base contains six terrain factor overlays: slope, vegetation, soils, hydrographic, transportation, and obstacles. This data base must be supplemented by current intelligence information in order to provide an up-to-date picture of the battlefield terrain. The terrain analysis should enable the commander to see the friendly and enemy courses of action and aid in making the best use of the terrain features in the area.
- Weather Analysis (third function). This analysis is crucial to certain operations where weather can determine avenues of approach or methods of transportation. The commander's aim in this analysis is to use the information to minimize the weather effects through planning. Weather analysis is also closely integrated with terrain since weather has such a tremendous effect on terrain. Rain can greatly affect avenues of approach and wind may affect helicopter or landing craft routes. Weather and terrain analysis are both represented in the IPB process by overlays. The graphics allow for the visual integration of areas of interest, weather and terrain.
- Threat Evaluation (fourth function). This consists of an in-depth study of the opposing forces to include their organization, tactical doctrine, weapons and equipment. The Marine Corps intelligence unit would determine the threat capabilities and how the threat might operate relative to their doctrine and training. They then produce "doctrinal templates" which depict the enemy doctrinal deployment and reproduce these to distribute to subordinate intelligence sections.
- Threat Integration (fifth function). Threat integration is the process of developing probable situations and events that might occur by utilizing the analysis from functions 1-4. Templates are created in this phase which might depict what a threat force could do at a specific time and

place on the battlefield. In this function critical areas become apparent and are designated as **named areas of interest** (NAIs). Also **target areas of interest** (TAIs) are identified.

The IPB process identifies areas on the battlefield where significant events or activities will take place. It also specifies targets that can be attacked to support the force commander's concept of operations and fire support plan. It does not, however, weight different options the commander may have when selecting the avenues of approach. It also does not take into account casualties.

The model described and implemented in the following sections takes the IPB process one step further. It takes into account weather and terrain, and also adds the element of resistance in order to minimize casualties in the case that the commander has optional routes of aggression. In the recent war against Iraq the casualty issue was very important. Casualty reduction will continue to be a future area of concern.

II. INTELLIGENCE MODEL

This chapter reviews the intelligence preparation taken by the Marine Corps, concentrating on the specific areas of the preparation which are utilized by the intelligence model developed here: the Marine Amphibious Landing Model. It discusses the methodology used in structuring the model and the mathematical formulation which solves the minimum cost transshipment problem.

A. GENERAL MODEL OVERVIEW

Intelligence Preparation of the Battlefield (IPB) is essentially separated into two phases: a pre-hostility and post-hostility phase. The model developed in this thesis is specifically oriented toward the pre-hostility phase or before a hostile advance is taken. Because of the flexibility of this type of model, minor changes could be made to orient it toward a post-hostility phase. The post-hostility phase (or after an attack has been initiated) would simply require a faster reaction time in collecting the necessary intelligence information to input into the model.

A review of the necessary intelligence information that is collected during the IPB process is as follows:

1. Battlefield area evaluation.
2. Terrain information to include obstacles, avenues of approach and mobility corridors.
3. Weather and its effect on terrain.
4. Enemy doctrinal templates (how the enemy fights) and event templates which depict how enemy formations would move through the battlefield.

5. Threat integration and the production of decision support templates which identify target areas of interest (TAIs) and specify where and when critical decisions need to be made.

The above five functions assist the commander in preparation of offensive operations. In every battle the commander is faced with a set of alternative choices as to the avenue of advance in order to achieve the objective(s). An avenue of advance is a route or path which is taken by the advancing Marines and will ultimately lead to the objective. With the intelligence information collected, a battle plan can be devised. The Marine Amphibious Landing Model is somewhat simplified in order to give the commander a clearer picture of the cost associated with each avenue of advance and also provide the specific avenues which would minimize the cost.

The Marine Amphibious Landing model utilizes information from the five functions in the IPB process as follows:

1. Predicted weather condition on the day of attack (function 3).
2. Estimated terrain (or sea state) condition for each avenue of advance (function 2).
3. Estimated resistance based upon information obtained in function 4.
4. Speed of movement along course of advance (function 1).
5. Distance to each phase of the advance (function 1).

The model is applied to an amphibious landing followed by a land advance to achieve specific objectives. With the information listed above, a cost is determined for each avenue of advance in order to allow comparisons of different offensive actions.

The cost determined in this model is a function of weather, terrain, resistance, speed and distance, and is a relative measure of casualties per time. Each of the five categories listed is necessary for the following reasons:

1. Weather affects surface movement of ships, landing craft and vehicles as well as air operations. The weather problems to be taken into account are winds which affect surf, tides, air operations and employment of chem/nuc weapons; reduced visibility due to fog, snow, and rain; and temperature which reduces operating efficiency of motorized equipment and can adversely affect personnel. Weather is taken into account in each phase of the assault.

2. Terrain not only affects troop movement on land but affects personnel landing on beaches. The "terrain" referred to here may be the swell of the sea or the surf which breaks as the landing craft approach shore. Other necessary information for amphibious landings include tidal periods, currents, sand bars, rocks, shoals and reefs. All of these could contribute adversely to an amphibious landing and need to be considered when analyzing the terrain.

3. Resistance plays a larger role in the cost function and thus is weighted accordingly. The smart tactician will choose the path of least resistance in order to save time and lives. Often an objective is assigned to a unit when no easy path to the objective is available. Intelligence estimates of the enemy must be relied upon. The estimates could be possible minefields, enemy bases of fire or known enemy positions.

4. Speed is a contributor to the cost function because speed can be a determining factor for the length of time a unit is required to stay in enemy territory or an enemy line of fire. Speed is also critical when time to achieve an objective is important. If the speed a unit can maneuver along an avenue adversely affects the mission either by time ineffectiveness or casualties then the cost function will reflect a higher value.

5. Distance of the avenue may or may not be a contributing factor to the success of the mission. The distance to an objective may be further but the speed may be faster and the resistance less, thus the avenue is safer. The distance along each avenue can be directly proportional to the amount of resistance encountered and is therefore a factor in this model.

There are many more factors which could be taken into account in determining a cost function for this model. These five were chosen based on the existing IPB process. Further research into a more intricate cost function could be the basis of future study.

B. METHODOLOGY

The model was designed in three stages. In the first stage a program was created to aid the user in the collection of intelligence information. This stage is interactive and requires a significant amount of input from the user. The program for the first stage is written in the FORTRAN programming language. Its basic functions are to collect data, assign real variables to the user's intelligence estimates and create a cost function table to be used in a future stage. The FORTRAN program, called "INTEL" is enclosed as Appendix A.

In the second stage a file is generated that will be used by the General Algebraic Modeling System (GAMS). This file is automatically created after the user finishes inputting data during stage 1. The file that is created, AMPH GAMS A1 (Appendix B), can be executed by itself or it can be placed in an executive file which will execute all stages of the model.

The third stage creates the output file for analysis by the intelligence and operations sections. The output file provides information such as the feasible avenues given the time constraint, the most optimal avenues of assault by the commander, and the number of Marines attrited. The output is described in more detail in the following chapter. See Figure 2 for a schematic of the model stages.

C. MATHEMATICAL FORMULATION

The model is formulated as a transshipment problem. The ships (source nodes) contain the number of Marines available to be transported via the beachheads (transshipment nodes) for the purpose of conducting an assault on the objectives (sink nodes). Each node has a requirement for a minimum number of Marines and there is a cost associated with each arc or avenue of approach.

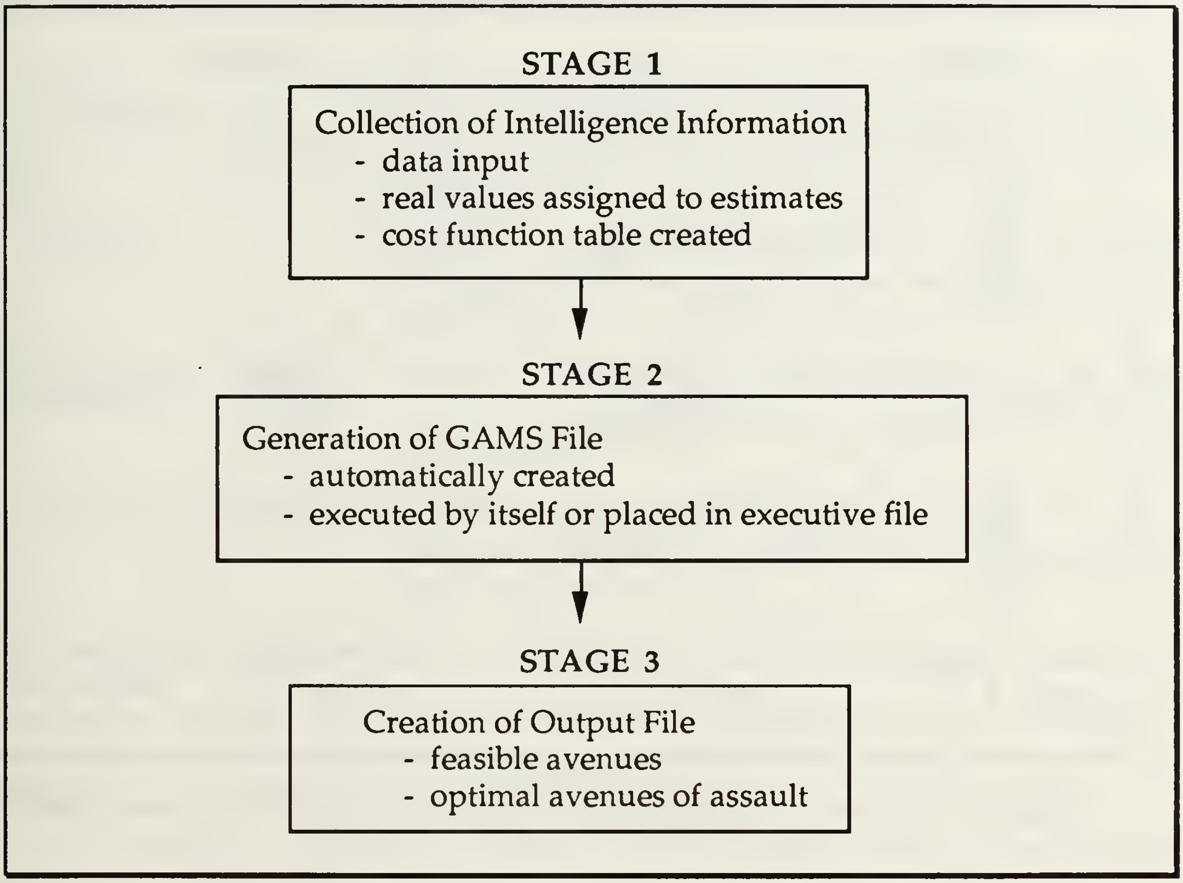
STAGE 1
Collection of Intelligence Information
- data input
- real values assigned to estimates
- cost function table created
STAGE 2
Generation of GAMS File
- automatically created
- executed by itself or placed in executive file
STAGE 3
Creation of Output File
- feasible avenues
- optimal avenues of assault

Figure 2. Schematic of Model Stages

The goal of the network is to find that (those) avenue(s) that will minimize the total arc cost. The cost is in terms of Marines lost (casualties). The underlying premise behind this approach is that the greater the resistance, the further the distance, and the more severe the conditions of travel, the higher the risk of casualties. Casualties may be the result of actual loss of life or Marines detained at a location due to the strength of resistance or impassibility of an avenue. See Figure 3 for an example of the transshipment network.

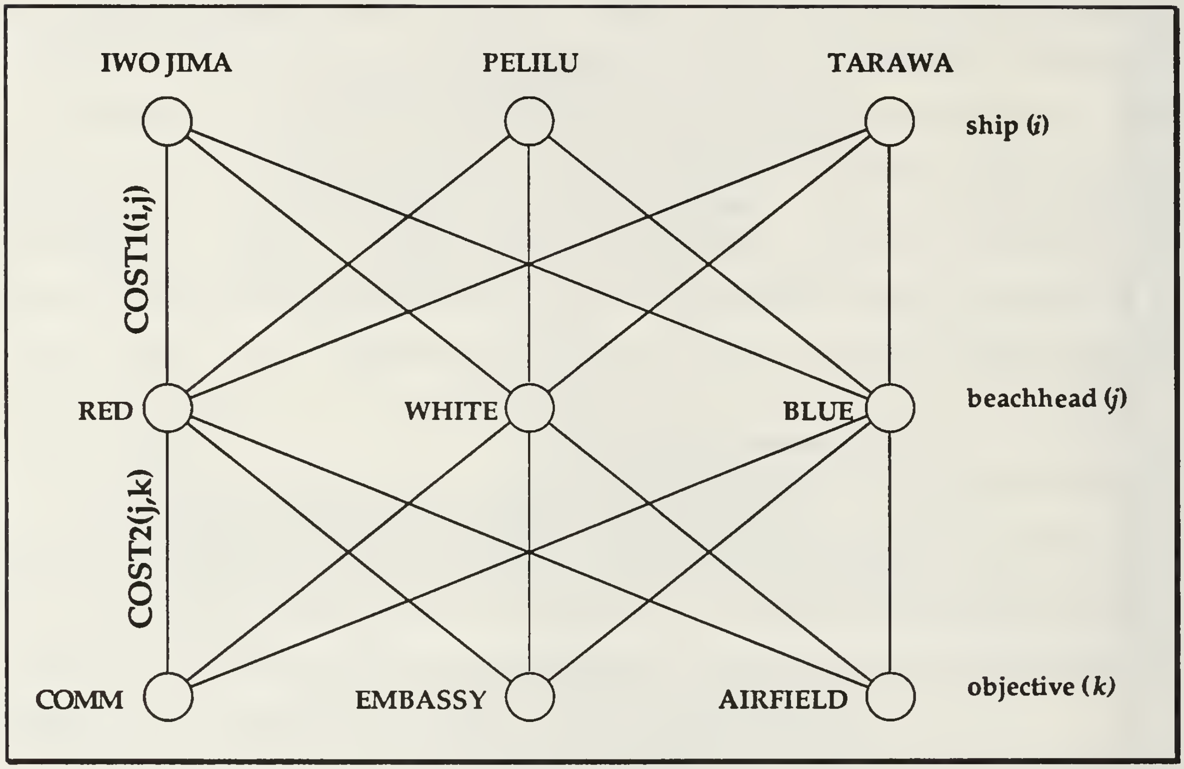

Figure 3. Network Presentation of the Marine Corps Amphibious Model

Once all nominal values are assigned numerical values in the main program, a "casualty" rate table is created. The casualty rate value is actually a proportion rate, providing the percentage of a force that will be eliminated per unit time on an avenue. These values are utilized in the GAMS program for the arc cost calculation. The casualty rate values for this table are based upon the following equations:

$$\mathrm{CAS1}_{ij} = \mathrm{RES1}_{ij} + \omega\left(\mathrm{WX1}_{ij} + \mathrm{TERR1}_{ij}\right) \qquad (2.1)$$

$$\mathrm{CAS2}_{jk} = \mathrm{RES2}_{jk} + \omega\left(\mathrm{WX2}_{jk} + \mathrm{TERR2}_{jk}\right) \qquad (2.2)$$

where

$CAS1_{ij}$ = casualty rate value from ship (i) to beachhead (j).

$CAS2_{jk}$ = casualty rate value from beachhead (j) to objective (k).

$RES1_{ij}$ = resistance numerical value for the nominal estimate (low, med, high) from ship (i) to beachhead (j).

$RES2_{jk}$ = resistance numerical value for the nominal estimate (low, med, high) from beachhead (j) to objective (k).

$WX1_{ij}$ and $TERR1_{ij}$ = weather and terrain numerical values for the nominal estimates (fair, med, bad) from ship (i) to beachhead (j).

$WX2_{ij}$ and $TERR2_{ij}$ = weather and terrain numerical values for the nominal estimates (fair, med, bad) from beachhead (j) to objective (k).

With the consideration that resistance will be the major cause of casualties, weather and terrain are presently weighted less in the casualty equations. In equations 2.1 and 2.2 the resistance is weighted ω times the value of weather and terrain estimates. This weighting value can be adjusted based on the situation and at the commander's discretion.

The casualty costs are used in the GAMS program to determine the overall cost for each arc. The cost functions result in the following:

$$COST1_{ij} = CAS1_{ij} * DIST1_{ij} \div SPEED1_{ij} \tag{2.3}$$

$$COST2_{jk} = CAS2_{jk} * DIST2_{jk} \div SPEED2_{jk} \tag{2.4}$$

As can be seen by equations (2.3) and (2.4) the costs are functions of estimated casualty rates, distance and speed. They are actually the percentage of a force that will be lost during the time the force is on a specific avenue. This is based upon the premise that an offensive unit which will be traveling further and at a slower speed will more likely be delayed and/or suffer greater casualties.

The intelligence process can then be modeled using the above costs as a minimum cost transshipment problem. The formulation is:

$$\text{minimize} \sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\left(\left(\mathrm{COST1}_{ij}\right)*\left(X_{ijk}+S_{ijk}\right)\right)+\sum_{j\in J}\sum_{k\in K}\left(\mathrm{COST2}_{jk}*Y_{jk}\right) \qquad (2.5)$$

subject to

$$\sum_{j\in J}\sum_{k\in K}X_{ijk}\le A_i, \quad \text{all } i\in I \qquad (2.6)$$

$$\sum_{i\in I}\sum_{k\in K}\left(\left(1-\mathrm{COST1}_{ij}\right)*\left(X_{ijk}+S_{ijk}\right)\right)\ge B_j, \quad \text{all } j\in J \qquad (2.7)$$

$$\sum_{j\in J}\left(Y_{jk}-\left(\mathrm{COST2}_{jk}*Y_{jk}\right)\right)=C_K, \quad \text{all } k\in K \qquad (2.8)$$

$$\sum_{k\in K}Y_{jk}\le\sum_{i\in I}\sum_{k\in K}\left(\left(1-\mathrm{COST1}_{ij}\right)*\left(X_{ijk}+S_{ijk}\right)\right) \qquad (2.9)$$

$$\text{all } j\in J$$

$$Y_{jk}\le\sum_{i\in \varepsilon}\left(\left(1-\mathrm{COST1}_{ij}\right)*\left(X_{ijk}+S_{ijk}\right)\right)$$

$$\text{all } j,k\in JK \qquad (2.10)$$

$$E-\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}S_{ijk}\le 0 \qquad (2.11)$$

$$100000*E-\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}S_{ijk}\ge 0 \qquad (2.12)$$

$$\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}X_{ijk}-\left(\sum_{i\in I}A_i\right)*E\ge 0 \qquad (2.13)$$

$$\left(\left(\mathrm{DIST1}_{ij} \div \mathrm{SPEED1}_{ij}\right) + \left(\mathrm{DIST2}_{jk} \div \mathrm{SPEED2}_{jk}\right)\right) * T_{ijk} \le \mathrm{MAXTIME}_{k}, \quad \text{all } (ijk) \in (IJK) \tag{2.14}$$

$$-1 * \left(\mathrm{DIST1}_{ij} \div \mathrm{SPEED1}_{ij}\right) + \left(\mathrm{DIST2}_{jk} \div \mathrm{SPEED2}_{jk}\right) \le \left(T_{ijk} - 1\right) * \mathrm{MAXTIME}_{k}, \quad \text{all } (ijk) \in (IJK) \tag{2.15}$$

$$\left(X_{ijk}\right) \le 100000 * T_{ijk}, \quad \text{all } (ijk) \in (IJK) \tag{2.16}$$

$$X_{ijk} \ge 0, \ \text{all } (i,j,k) \in (I,J,K) \tag{2.17}$$

$$S_{ijk} \ge 0, \ \text{all } (i,j,k) \in (I,J,K) \tag{2.18}$$

$$E, T_{ijk} = 0,1 \quad \text{all } (i,j,k) \in (I,J,K) \tag{2.19}$$

where

I = set of all ships,

J = set of all beachheads,

K = set of all objectives,

X_{ijk} = Marines available to go from ship (i) to objective (k) via beachhead (j) (variable),

Y_{jk} = Marines going from beachhead (j) to objective (k),

$\mathrm{COST1}_{ij}$ = cost of flow from i to j,

$\mathrm{COST2}_{jk}$ = cost of flow from j to k,

A_i = number of Marines available on ship i (constant),

B_i = number of Marines needed at beachhead j (constant),

C_k = number of Marines needed at objective k (constant),

ε = The set of marines originating at ship i allocated for use at objective k via beachhead j,

S_{ijk} = elastic variable,

E = binary variable,

T_{ijk} = binary variable,

MAXTIME$_k$ = maximum time allowed for Marines to achieve objective k (constant).

The set of constraint equations actually encompasses three separate major requirements. Equations 2.6–2.9 are the resource constraints. Equation (2.6) ensures that the total number of Marines leaving ship (i) will be less than or equal to the number of Marines available on ship (i). A_i, the number of Marines available, in equation (2.6) is input by the user as are all constants in the formulation. Equation (2.7) ensures that the requirements for Marines on beachhead (j) is met. It takes into account the Marines that are lost going from their ship to the designated beachhead. The last two equations (2.8–2.9) ensure that the requirement for Marines at objective (k) is met.

Constraint Equation 2.10 is generated for every arc (i,j,k) which meets the time constraint (maxtime$_k$). It ensures the number of Marines traveling from beachhead (j) to objective (k) is less than the total number of Marines arriving at beachhead (j).

Equations 2.11 through 2.13 are sufficiency requirements. They control the elastic varible (S_{ijk}) which is located in the resource constraints. The elastic variable allows additional Marines to be added if it is necessary to successfully complete the mission. The additional Marines are under the same restrictions as those Marines departing the ship and therefore suffer losses. The Marines that enter the network by the elastic variable allow the model to obtain a feasible solution. The output generated will show the S_{ijk} values and should be interpreted as the number of Marines that are lacking on ship (i) in the force commander's plan to properly execute the mission. If

S_{ijk} then the elastic variable would not be needed and S_{ijk} would maintain a value of zero.

The following discussion demonstrates how the elastic variable is applied. If $\sum_{i\in I}\sum_{j\in J}\sum_{k\in K} S_{ijk} > 0$ then extra Marines are needed. It is necessary for the Marines currently in the network to be used prior to obtaining extra Marines. Thus the constraint

$$\sum_{i\in I}\sum_{j\in J}\sum_{k\in K} X_{ijk} = \Sigma A_i \tag{2.20}$$

is satisfied. However, if there are sufficient Marines in the network then $\sum_{i\in I}\sum_{j\in J}\sum_{k\in K} S_{ijk} = 0$ and

$$\sum_{j\in J}\sum_{k\in K} X_{ijk} \le \sum_{i\in I} A_i. \tag{2.21}$$

This leads to the binary variable "E" signifying elasticity and being defined as follows:

$$E = \begin{cases} 1 & \text{if } \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} S_{ijk} > 0 \\ 0 & \text{if } \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} S_{ijk} = 0. \end{cases} \tag{2.22}$$

This is accomplished by constraint equations (2.11–2.12). Finally, in order to force the use of all Marines prior to obtaining extra Marines from another source, the constraint 2.13 is applied.

The final major requirement in the constraint equations is the time factor. If the objectives have to be taken within a specific time, then only those avenues which allow transit within that time will be allowed. Equations 2.14 and 2.15 ensure the transit time for Marines going along avenues i to k is under the maximum time allowed for that objective. If the avenue is not feasible, equation 2.16 forces the Marines along another feasible avenue.

III. IMPLEMENTATION OF THE MARINE AMPHIBIOUS LANDING MODEL

This chapter discusses the input and output of the Marine Amphibious Landing Model. The input that is required to run the program is both nominal, in the case of the intelligence estimates, and interval level data. The output will give the commanding officer and the intelligence officer information such as the feasible avenues with the given time constraint and the optimal number of Marines on each avenue. The output additionally informs the commander if additional Marines are required and on which ship these Marines need to be located.

A. MODEL INPUT

The input necessary to run the program is lengthy, therefore a format for information collection and database management has been developed and can be found in Appendix C. Each section of the database form is what the user will see at each input prompt in the program.

The model, called the Marine Amphibious Landing Model, is set up as a three-tiered network. It assumes that there will be one or more ships (first tier) which will contain Marines to be off-loaded (either by helicopter or landing craft) for the amphibious landing. The second tier in the network is the location on the shore (beachhead) where the Marines will make an initial landing. The last tier is the objective or set of objectives which will be seized following the landing (See Figure 2 for network diagram).

The first three prompts set up the network. The first prompt will ask for the number of ships carrying Marines and then the names of the ships. The names of the ships that are used must be six or fewer characters and each entry must be followed by a carriage return. The six or fewer characters is a requirement for all future inputs. The next prompt will ask for the number of landing zones and their names. The names of the ships and landing zones can be considered "code names" and may be as simplistic as "red," "white" and "blue." The third prompt is the number of objectives and the names of those objectives. The program will only allow the user to enter the number of names that was entered in the previous number response. Therefore, if the user entered "3" when asked for the number of objectives, then only "3" names will be permitted to be entered.

The next three prompts are necessary to execute the allocation portion of this model. The first prompt requests the number of Marines on each ship that was previously input. This number is the total number of Marines available on each ship that will carry out the mission from the ship to the beachheads and objectives. The next prompt requests the number of Marines needed at each beachhead. This may be the number of Marines required to take a defended beachhead or the number of Marines the commander feels is necessary to achieve further objectives. The third prompt is the commander's best estimate as to the number of Marines that will be required to seize each of the specific objectives. At this point all the information concerning the nodes of the network has been collected. The next set of prompts request information that will be used in determining the cost function along each arc of the network.

The weather, as stated before, can affect air and ground operations. All weather effects must be taken into consideration. Recall that weather can be viewed as gusty winds and temperature as well as fog, rain and snow. The user is asked at this prompt to give the best estimate of weather for each avenue of the assault. The choice of input is fair, medium or bad (fair, med, bad) to best describe the condition of the weather. If the avenue is not a consideration the user is to enter "none" at the prompt. The network will have one less arc in this case and the avenue will be taken entirely out of the program. The weather estimate input will be required at both tiers of the network; from the ships to the beachheads and the beachheads to the objectives.

The next input data is the resistance estimate. Resistance is the amount and intensity of enemy forces and/or obstacles that will be encountered based upon the intelligence estimate. Elements to be considered in this estimate are enemy strong points, obstacles such as trenches filled with burning oil, and mines both sea and land. Again the user will be asked to enter a subjective estimate: low, med or high.

Distance and speed between each node in the network are requested next. The distances between ship, beachhead and objective should be measured in *miles*. The speed of the landing craft, air craft or vehicle should be measured in *miles per hour*. These values are important in that they will determine the length of time the Marines will be in transit along an avenue.

The last input prompt collects the best estimate of the terrain or sea state; good, medium or difficult (good, med, dif). This estimate takes into account terrain features as discussed previously. Appendix D shows an echo of the input that was entered during one run of the program.

B. MODEL OUTPUT

The model output, produced upon GAMS execution, will display the following information:

1. The optimal number of Marines to send from ship (*i*) and beachhead (*j*).
2. The number of extra Marines that are needed on a particular avenue to successfully complete the mission.
3. The avenues of approach (arcs) which do not satisfy the constraint for the maximum time allowed.
4. The cost (or casualties) on each arc.

The above will each be considered in terms of GAMS output and an example of each are included in appendices E, F, and G. In each of the tables, a blank represents a zero.

The first, the optimal number of Marines to send from ship (*i*) and beachhead (*j*), is listed in Appendix E under Parameter Summary1 and Parameter Summary2. The left hand column shows where the Marines will be sent from while the row across the top displays where they will be sent to. In this example, there are approximately 130 Marines going from the Iwo Jima to the Red beachhead and 161 Marines going from the White beachhead to objective Comm.

The second parameter that is shown in the output is under Parameter Summary3 (see Appendix F). This table shows the number of extra Marines that are needed at a particular ship in order to successfully complete the mission. In the example in Appendix F the Pelilu required 59 more Marines going to beachhead White and approximately 325 more Marines going to beachhead Blue in order to feasibly complete the mission. Thus if 384 additional Marines are placed on the Pelilu the solution will only require additional Marines on the Iwo Jima.

The third variable to be displayed is the time variable T.L. It is again displayed in table format. Since the variable T is a binary variable only "1.000" or a blank will be shown. Appendix F is an example where 15 avenues (or arcs) do not meet the requirement for time and are represented in the table by blanks (Pelilu.Blue, Tarawa.White, and Tarawa.Blue are not displayed in the table because all routes along these avenues have a zero value). Appendix G is a solution where every avenue meets the time constraint.

The user can determine the cost (casualties) that are attributed to each avenue by looking under the Parameter Summaries. In the example in Appendix G there are just over seven Marines lost that leave the ship Iwo Jima, almost 13 Marines lost leaving beachhead Red, and almost four casualties from the additional Marines added to the Pelilu.

Appendix H is a sensitivity analysis of the number of casualties. From Appendix G each of the casualties rates were increased by .01. This resulted in increased casualties along almost every route. The optimization procedure in the program reroutes Marines to avenues which will result in fewer casualties and this is the reason the casualty numbers decrease in some instances. Due to the increased casualty rates the number of additional Marines required (Parameter Summary3) has also increased to a total of 314 from the previous 285.

IV. CONCLUSION

Based on the current process known as IPB, this thesis has provided an analytical method to assist the commander in the decision-making prior to an amphibious assault. The information that is provided as output will tell the commander where to place Marines, how many extra Marines may be required, casualty estimates and avenues of aggression which meet the time constraint. The thesis is to be used as an aid and could never completely orchestrate a battle and replace the instinctive perceptivity of the battlefield commander.

The thesis has been written with the intention of not only providing an analytical tool but also to aid the association between the intelligence and operations sections. The difficulties in communicating between the two sections must be alleviated in order for the battle plan to progress in a smooth and coherent manner. The model is designed in a way that requires input from both sections. Written with the intention of having the intelligence officer as the "user," the model requires the "user" to gain knowledge of the battle plan prior to implementing the Amphibious Landing Program. This will impel the two sections to work more closely together.

The model was designed with flexibility in mind. Every battlefield is unique and requires flexibility in order to be adaptable on every occasion. This also is not a model which simulates all of the events on the battlefield and thus lends itself to future enhancements and additional provisions. The remaining pages of this thesis will discuss flexibility and ideas on future topics.

A. MODEL FLEXIBILITY

The model is flexible in that it can readily be adaptable to a different scenario other than the amphibious landing. It is currently designed in a three-tiered assault phase, however, with minor adjustments, could be expanded to a four or five-tiered model. The input would remain the same and more levels of objectives could be added. The model has been designed to allow a maximum of ten ships, ten beachheads, and ten objectives but this can also be easily altered by adjusting the parameter IP in the beginning of the FORTRAN code.

Within the model are various weighted values. The nominal inputs of weather, terrain and resistance are assigned a value by the program of .01, .02 and .03. These values will eventually reflect an increasing arc cost in the cost equations. The user may determine that these values are not appropriate and may desire to increase the cost more rapidly such as .01, .05, and 0.1.

Another weighted value in the model is in the casualty rate value equations 2.1 and 2.2. In these the weather and terrain have been weighted less than the resistance in determining a casualty value. This weighting can be adjusted accordingly if, for instance, the resistance is light but weather and terrain are severe. A possible weighting value might be 1.5 in this example. Another method of determining a casualty value is to have straight addition across all of the factors realizing resistance may play a larger role in the attrition of Marines.

B. FUTURE ENHANCEMENTS

The model in this thesis is a stand-alone intelligence model. The battlefield is a multifaceted and dynamic scenario. Many areas such as supply/logistics, naval gunfire support, air support and demolitions could be modeled and implemented in conjunction with the intelligence model to make the battlefield analysis more all-encompassing.

Another area of future study could be on a more intricate cost function. The cost function in this thesis is based on five aspects of the battlefield. Other factors that can influence casualties or delays may be associated with unit training level or unit motivation, attitude, and morale. There are other areas as well that can be researched and play a role in the cost function.

Since each battlefield may require other network scenarios, a more flexible model could be developed to handle such situations. The possibility exists that a beachhead may also be an objective. In this network scenario there would be no arcs leading from the beachhead. The model could be enhanced to allow for this by adjusting the user input and setting the distance for all arcs from the beachhead to zero.

A final enhancement would be to expand the model to give the commander the option of sending single Marines or units of Marines such as platoons or battalion landing teams. This may be a more realistic option because generally a commander will not send 3 or 4 Marines along an avenue but a group of Marines to maintain unit cohesion.

APPENDIX A. FORTRAN PROGRAM "INTEL"

```
FILE: INTEL    FORTRAN  A1

        PROGRAM MARINE

***********************************************************************
*   THIS IS AN INTERACTIVE PROGRAM WHICH WILL GENERATE THE DATA       *
*   NEEDED TO RUN A GAMS OPTIMIZATION PROGRAM.  THE PURPOSE OF        *
*   THIS EXERCISE IS TO AID THE INTELLIGENCE OFFICER WITH             *
*   GATHERING AND INTERPRETING INFORMATION BASED ON INTELLIGENCE      *
*   RESOURCES AND ALSO THE DESIRES OF THE OPERATIONS OFFICER.         *
***********************************************************************

      INTEGER IP
      PARAMETER (IP = 10)
      INTEGER ANUM1, ANUM2, ANUM3, I, J, K, M, P, Q, LEVEL1, LEVEL2
      INTEGER  OBJMAR(IP), SHIMAR(IP), LANMAR(IP)
      REAL DIST1(IP,IP), SPEED1(IP,IP), RES1(IP,IP), WX1(IP,IP)
      REAL TERR1(IP,IP), CAS1(IP,IP), DIST2(IP,IP), SPEED2(IP,IP)
      REAL RES2(IP,IP), WX2(IP,IP), TERR2(IP,IP), CAS2(IP,IP)
      REAL MAXTIME(IP)
      CHARACTER ANS, ANS1*4, ANS2*4, SHIP(IP)*6, LAND(IP)*6, OBJ(IP)*6

************************************************************************
* THE TOTAL NUMBER OF NODES FOR EACH LEVEL I'VE LIMITED TO IP, IE) IP *
* SHIPS, IP LANDING ZONES AND IP OBJECTIVES.  THE NUMBER OF CHARACTERS *
* IN EACH ELEMENT IS LIMITED TO 6, IE) NAMES CAN BE NO LONGER THAN 6  *
* CHARACTERS. IP IS AN ADJUSTABLE PARAMETER.                          *
************************************************************************

      CALL EXCMS('CLRSCRN')
      CALL EXCMS('FILEDEF 20 DISK AMPH GAMS A1')
*     INFORM THE USER OF THE TYPE OF DATA NEEDED TO BE INPUT
      PRINT 10
10    FORMAT (' THE INFORMATION THAT YOU WILL NEED TO INPUT ARE THE'
     C   ,/,' ESTIMATES OF WEATHER, RESISTANCE, TERRAIN, DISTANCE '
     C   ,/,' AND THE SPEED OF YOUR MOVEMENTS.  REFER TO THESIS FOR'
     C   ,/,' DETAILED INSTRUCTIONS (TOPIC "MODEL INPUT").')

*     PRINT 15
*15    FORMAT (/,' DO YOU WISH TO CONTINUE?  ENTER Y OR N')
*      READ *, ANS
*      IF (ANS.EQ.'N') GO TO 999

************************************************************************
*                    NODES OF THE NETWORK                              *
*  THE FOLLOWING ENTRIES ARE THE NODES NEEDED IN THE NETWORK.  IN THIS *
*  SCENARIO THE ENTRIES WILL BE THE SHIPS ON WHICH THE MARINES WILL    *
*  DEPART, THE BEACH AREAS WHERE THE MARINES WILL LAND AND THE OBJEC-  *
*  TIVES WHICH THE MARINES WILL AGGRESS.                               *
************************************************************************

      PRINT 19
19    FORMAT (//,' HOW MANY SHIPS ARE THERE?')
      READ *, ANUM1
      PRINT 20
20    FORMAT(/,' INPUT THE NAMES OF THE SHIPS ON WHICH THE MARINES WILL'
     C   /,' BE DEPARTING FROM - ABBREVIATE - 6 CHARACTERS, NO SPACES')
      READ (*,21)(SHIP(I), I=1,ANUM1)
21    FORMAT(A6)
      CALL EXCMS('CLRSCRN')
      PRINT 23
23    FORMAT (' HOW MANY BEACH LANDING ZONES ARE THERE?')
      READ *, ANUM2

      PRINT 25
25    FORMAT (/,' INPUT THE LANDING ZONES OR BEACHHEADS - 6 CHARACTERS',
     C         ' NO SPACES')
      READ (*,21)(LAND(J), J=1,ANUM2)

      CALL EXCMS('CLRSCRN')
      PRINT 29
29    FORMAT (' HOW MANY OBJECTIVES ARE THERE?')
      READ *, ANUM3




      PRINT 30
30    FORMAT (/,' INPUT THE OBJECTIVE NAMES - 6 CHARACTERS, NO SPACES')
      READ (*,21)(OBJ(K), K=1,ANUM3)

****THE FOLLOWING INPUTS WILL BE USED IN THE GAMS PROGRAM.  THE NUMBER
****OF MARINES NEEDED IN EACH PHASE WILL BE OPTIMIZED.

      CALL EXCMS('CLRSCRN')
      DO 35 M = 1, ANUM1
        PRINT 33, SHIP(M)
33      FORMAT(' INPUT THE NUMBER OF MARINES ON ',A)
        READ *, SHIMAR(M)
35    CONTINUE

      CALL EXCMS('CLRSCRN')
      DO 38 M = 1, ANUM2
        PRINT 36, LAND(M)
36      FORMAT(' INPUT THE NUMBER OF MARINES NEEDED ON ',A)
        READ *, LANMAR(M)
38    CONTINUE

      CALL EXCMS('CLRSCRN')
      DO 40 M=1,ANUM3
        PRINT 39, OBJ(M)
39     FORMAT(' INPUT THE NUMBER OF MARINES NEEDED TO ACHIEVE OBJ ',A)
        READ *, OBJMAR(M)
40    CONTINUE

      CALL EXCMS('CLRSCRN')
      DO 42 M=1,ANUM3
        PRINT 43, OBJ(M)
43     FORMAT(' INPUT THE TIME IN HOURS FROM H-HOUR TO ACHIEVE OBJ ',A)
        READ *,MAXTIME(M)
42    CONTINUE
*
************************************************************************
```

```
************  COLLECT WEATHER DATA  *************

      CALL EXCMS('CLRSCRN')
      PRINT 100
100   FORMAT (' INPUT THE BEST WEATHER ESTIMATE (FAIR,MED,BAD) FROM',
     C       /,' EACH SHIP TO EACH LANDING ZONE. INPUT "NONE" IF THE',
     C       /,' AVENUE WILL NOT BE CONSIDERED')

****INPUT WX DATA FROM SHIP TO SHORE

      DO 115 LEVEL1 = 1,ANUM1
           DO 110 LEVEL2 = 1,ANUM2
             PRINT 105, SHIP(LEVEL1), LAND(LEVEL2)
105          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS1
107          FORMAT(A4)

108          IF (ANS1.NE.'FAIR'.AND.ANS1.NE.'MED '.AND.ANS1.NE.'BAD '.
     C           AND.ANS1.NE.'NONE')  THEN
               PRINT 109
109            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-FAIR,MED,BAD,NONE')
               READ (*,107) ANS1
               GO TO 108
             ENDIF

               IF (ANS1.EQ.'FAIR') THEN
                   WX1(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS1.EQ.'MED') THEN
                   WX1(LEVEL1,LEVEL2) = 0.02
                 ELSEIF (ANS1.EQ.'BAD') THEN
                   WX1(LEVEL1,LEVEL2) = 0.03
                 ELSE
                   WX1(LEVEL1,LEVEL2) = 10000.0


               ENDIF
********THE LAST ELSE WILL BE EXECUTED WHEN 'NONE' HAS BEEN SELECTED***

110        CONTINUE
115   CONTINUE
      CALL EXCMS('CLRSCRN')
      PRINT 200
200   FORMAT (' INPUT THE BEST WEATHER ESTIMATE (FAIR,MED,BAD) FROM'
     C      /, ' EACH LANDING ZONE TO EACH OBJ. INPUT "NONE" IF THE'
     C      /,' AVENUE WILL NOT BE CONSIDERED')

*****INPUT WX DATA FROM SHORE TO OBJ

      DO 215 LEVEL1 = 1,ANUM2
           DO 210 LEVEL2 = 1,ANUM3
             PRINT 205, LAND(LEVEL1), OBJ(LEVEL2)
205          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS2
206          IF (ANS2.NE.'FAIR'.AND.ANS2.NE.'MED '.AND.ANS2.NE.'BAD '.
     C           AND.ANS2.NE.'NONE')  THEN
               PRINT 207
207            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-FAIR,MED,BAD,NONE')
               READ (*,107) ANS2
               GO TO 206
             ENDIF
```

```
               IF (ANS2.EQ.'FAIR') THEN
                   WX2(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS2.EQ.'MED') THEN
                   WX2(LEVEL1,LEVEL2) = 0.02
                 ELSEIF (ANS2.EQ.'BAD') THEN
                   WX2(LEVEL1,LEVEL2) = 0.03
                 ELSE
                   WX2(LEVEL1,LEVEL2) = 10000.0
               ENDIF
210        CONTINUE
215   CONTINUE




************  COLLECT RESISTANCE DATA  ****************

      CALL EXCMS('CLRSCRN')
      PRINT 320
320   FORMAT (' INPUT THE BEST ESTIMATE FOR RESISTANCE (LOW,MED,HIGH)',
     C        /,' FROM SHIP TO EACH LANDING ZONE.')


      DO 335 LEVEL1 = 1,ANUM1
           DO 330 LEVEL2 = 1,ANUM2
             IF (WX1(LEVEL1,LEVEL2).GE.9999.0) GO TO 330

             PRINT 325, SHIP(LEVEL1), LAND(LEVEL2)
325          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS2
326          IF (ANS2.NE.'LOW '.AND.ANS2.NE.'MED '.AND.ANS2.NE.'HIGH')
     C          THEN
               PRINT 327
327            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-LOW,MED,HIGH')
               READ (*,107) ANS2
               GO TO 326
             ENDIF
               IF (ANS2.EQ.'LOW') THEN
                   RES1(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS2.EQ.'MED') THEN
                   RES1(LEVEL1,LEVEL2) = 0.02
                 ELSE
                   RES1(LEVEL1,LEVEL2) = 0.03
               ENDIF
330        CONTINUE
335   CONTINUE
      CALL EXCMS('CLRSCRN')
      PRINT 320
350   FORMAT (' INPUT THE BEST ESTIMATE FOR RESISTANCE (LOW,MED,HIGH)',

     C        /,' FROM LANDING ZONE TO EACH OBJ. ')


      DO 365 LEVEL1 = 1,ANUM2
           DO 360 LEVEL2 = 1,ANUM3
             IF (WX2(LEVEL1,LEVEL2).GE.9999.0) GO TO 360
             PRINT 355, LAND(LEVEL1), OBJ(LEVEL2)
355          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS1
```

```
356          IF (ANS1.NE.'LOW '.AND.ANS1.NE.'MED '.AND.ANS1.NE.'HIGH')
     C           THEN
               PRINT 357
357            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-LOW,MED,HIGH')
               READ (*,107) ANS1
               GO TO 356
             ENDIF
               IF (ANS1.EQ.'LOW') THEN
                   RES2(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS1.EQ.'MED') THEN
                   RES2(LEVEL1,LEVEL2) = 0.03
                 ELSE
                   RES2(LEVEL1,LEVEL2) = 0.05
               ENDIF
360        CONTINUE
365   CONTINUE
```

```
******************  COLLECT DISTANCE DATA ******************

      CALL EXCMS('CLRSCRN')
      PRINT 400
400   FORMAT (' INPUT THE DISTANCES FROM EACH SHIP TO EACH LANDING',
     C   /,' ZONE. DISTANCES SHOULD BE IN MILES WITH ACCURACY AT MOST',
     C   /,' TO TWO DIGITS.  EXAMPLE - 2.10 ')

      DO 425 LEVEL1 = 1,ANUM1
           DO 415 LEVEL2 = 1,ANUM2
             IF (WX1(LEVEL1,LEVEL2).GE.9999.0) THEN
                 DIST1(LEVEL1,LEVEL2) = 100000.0
                 GO TO 415
             ENDIF
             PRINT 410,  SHIP(LEVEL1), LAND(LEVEL2)
410          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ *, DIST1(LEVEL1,LEVEL2)
415        CONTINUE
425   CONTINUE

      CALL EXCMS('CLRSCRN')
      PRINT 440
440   FORMAT (' INPUT THE DISTANCES FROM EACH LANDING ZONE TO EACH',
     C     /,' OBJ. DISTANCES SHOULD BE IN MILES WITH ACCURACY AT MOST'
     C    ,/,' TO TWO DIGITS.  EXAMPLE - 2.10 ')

      DO 475 LEVEL1 = 1,ANUM2
           DO 465 LEVEL2 = 1,ANUM3
             IF (WX2(LEVEL1,LEVEL2).GE.9999.0) THEN
                 DIST2(LEVEL1,LEVEL2) = 1.0
                 GO TO 465
             ENDIF
             PRINT 460, LAND(LEVEL1), OBJ(LEVEL2)
460          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ *, DIST2(LEVEL1,LEVEL2)
465        CONTINUE
475   CONTINUE
```

```
**************  COLLECT SPEED DATA  **************

      CALL EXCMS('CLRSCRN')
      PRINT 490
490   FORMAT (' INPUT THE SPEED FROM EACH SHIP TO EACH LANDING ZONE.',
     C     /,' SPEED SHOULD BE IN MILES PER HOUR WITH THE ACCURACY AT',
     C     /,' MOST TWO DIGITS EXAMPLE - 2.10 ')




      DO 505 LEVEL1 = 1,ANUM1
           DO 500 LEVEL2 = 1,ANUM2
             IF (WX1(LEVEL1,LEVEL2).GE.9999.0) THEN
                SPEED1(LEVEL1,LEVEL2) = 1.0
                GO TO 500
             ENDIF
             PRINT 495, SHIP(LEVEL1), LAND(LEVEL2)
495          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ *, SPEED1(LEVEL1,LEVEL2)
500        CONTINUE
505   CONTINUE

      CALL EXCMS('CLRSCRN')
      PRINT 510
510   FORMAT (' INPUT THE SPEED FROM EACH LANDING ZONE TO EACH OBJ.',
     C       /,' SPEED SHOULD BE IN MILES PER HOUR WITH THE ACCURACY AT',
     C         /,' MOST TWO DIGITS EXAMPLE - 2.10.')


      DO 525 LEVEL1 = 1,ANUM2
           DO 520 LEVEL2 = 1,ANUM3
             IF (WX2(LEVEL1,LEVEL2).GE.9999.0) THEN
                 SPEED2(LEVEL1,LEVEL2) = 1.0
                 GO TO 520
             ENDIF
             PRINT 515, LAND(LEVEL1), OBJ(LEVEL2)
515          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ *, SPEED2(LEVEL1,LEVEL2)
520        CONTINUE
525   CONTINUE

************* COLLECT TERRAIN DATA  ************

      CALL EXCMS('CLRSCRN')
      PRINT 615
615   FORMAT (' INPUT THE BEST TERRAIN (OR SEA STATE) ESTIMATE FROM',
     C    /,' EACH SHIP TO EACH LANDING ZONE  (GOOD,MED,DIF).')


      DO 635 LEVEL1 = 1,ANUM1
           DO 630 LEVEL2 = 1,ANUM2
             IF (WX1(LEVEL1,LEVEL2).GE.9999.0) GO TO 630
             PRINT 625, SHIP(LEVEL1), LAND(LEVEL2)
625          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS1
626          IF (ANS1.NE.'GOOD'.AND.ANS1.NE.'MED '.AND.ANS1.NE.'DIF ')
     C            THEN
               PRINT 627
627            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-GOOD,MED,DIF')
               READ (*,107) ANS1
               GO TO 626
             ENDIF
```

```
               IF (ANS1.EQ.'GOOD') THEN
                   TERR1(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS1.EQ.'MED') THEN
                   TERR1(LEVEL1,LEVEL2) = 0.02
                 ELSE
                   TERR1(LEVEL1,LEVEL2) = 0.03
               ENDIF
630        CONTINUE
635   CONTINUE


      CALL EXCMS('CLRSCRN')
      PRINT 645
645   FORMAT (' INPUT THE BEST TERRAIN (OR SEA STATE) ESTIMATE',
     C        /,' (GOOD,MED,DIF) FROM EACH LANDING ZONE TO EACH OBJ.')

      DO 665 LEVEL1 = 1,ANUM2


           DO 655 LEVEL2 = 1,ANUM3
             IF (WX2(LEVEL1,LEVEL2).GE.9999.0) GO TO 655
             PRINT 650, LAND(LEVEL1), OBJ(LEVEL2)
650          FORMAT (1X,'FROM ',A,1X,'TO ',A)
             READ (*,107) ANS1
651          IF (ANS1.NE.'GOOD'.AND.ANS1.NE.'MED '.AND.ANS1.NE.'DIF ')
     C           THEN
               PRINT 652
652            FORMAT (1X,'INCORRECT ENTRY,TRY AGAIN-GOOD,MED,DIF')
               READ (*,107) ANS1
               GO TO 651
             ENDIF
               IF (ANS1.EQ.'GOOD') THEN
                   TERR2(LEVEL1,LEVEL2) = 0.01
                 ELSEIF (ANS1.EQ.'MED') THEN
                   TERR2(LEVEL1,LEVEL2) = 0.02
                 ELSE
                   TERR2(LEVEL1,LEVEL2) = 0.03
               ENDIF
655        CONTINUE
665   CONTINUE




****CREATE CASUALTY TABLE . CAS(RES,WX,TERR)
***THIS TABLE WILL BE USED IN THE GAMS NETWORK FOR ARC COST CALCULATION

       DO 775 I=1,ANUM1
          DO 770 J=1,ANUM2
           IF (WX1(I,J).GE.9999.0) THEN
               CAS1(I,J) = 100000.0
               GO TO 770
           ENDIF
           CAS1(I,J) = RES1(I,J) + .1*(WX1(I,J)+TERR1(I,J))

 770      CONTINUE
 775   CONTINUE
```

```
       DO 785 J=1,ANUM2
          DO 780 K=1,ANUM3

           IF (WX2(I,J).GE.9999.0) THEN
               CAS2(I,J) = 100000.0
               GO TO 780
           ENDIF
           CAS2(J,K) = RES2(J,K) + .1*(WX2(J,K)+TERR2(J,K))

 780      CONTINUE
 785   CONTINUE




************************************************************************
*GENERATE FILE FOR GAMS
************************************************************************

      WRITE(20,800)
800   FORMAT('$TITLE MARINE AMPHIB LANDING MODEL')
      WRITE(20,805)
805   FORMAT('$STITLE THESIS BY CATHY JOHNSON')

      WRITE(20,810)
810   FORMAT('$OFFSYMXREF OFFSYMLIST OFFUPPER')
      WRITE(20,815)
815   FORMAT(1X,'OPTIONS LIMCOL = 0, LIMROW = 0, SOLPRINT = OFF ;',/,
     C      1X,'OPTIONS OPTCR = 0.0001 ;')
      WRITE(20,*) 'SETS'
      WRITE(20,*) '    I  SHIPS OFF SHORE / '
         DO 825 L=1,ANUM1
            IF (L.EQ.ANUM1) THEN
                WRITE(20,*) SHIP(L)


              ELSE
                WRITE(20,*)  SHIP(L), ','
            ENDIF
825      CONTINUE
      WRITE(20,*) ' /  '

      WRITE(20,*)  '  J  LANDING BEACHES / '
         DO 835 L=1,ANUM2
            IF (L.EQ.ANUM2) THEN
               WRITE(20,*) LAND(L)
              ELSE
               WRITE(20,*)  LAND(L), ','
            ENDIF
835      CONTINUE
      WRITE(20,*) '/ '

      WRITE(20,*) '  K  OBJECTIVES / '
        DO 845 L=1,ANUM3
            IF (L.EQ.ANUM3) THEN
               WRITE(20,*) OBJ(L)
              ELSE
               WRITE(20,*) OBJ(L), ','
            ENDIF
845     CONTINUE
```

```
      WRITE(20,*) '/ ;'
      WRITE(20,*) 'PARAMETERS'
      WRITE(20,*) 'A(I)  NUMBER OF MARINES ON SHIP I  /'
      DO 850, N=1,ANUM1
          WRITE(20,*) SHIP(N), SHIMAR(N)
850   CONTINUE
      WRITE(20,*) '/ '
      WRITE(20,*) 'B(J) NUMBER OF MARINES NEEDED AT BEACHHEAD J /'
        DO 855, N=1,ANUM2
          WRITE(20,*) LAND(N), LANMAR(N)
855     CONTINUE
      WRITE(20,*) '/ '
      WRITE(20,*) 'C(K) # OF MARINES REQUIRED TO COMPLETE MISSION /'
        DO 860, N=1,ANUM3
           WRITE(20,*) OBJ(N), OBJMAR(N)
860     CONTINUE
      WRITE(20,*) '/ ;'

      WRITE(20,*) 'PARAMETER DIST1(I,J) DISTANCE FROM SHIP TO BEACH  /'
        DO 870, P=1,ANUM1
            DO 865, Q=1,ANUM2
                 WRITE(20,*) SHIP(P),'.',LAND(Q), DIST1(P,Q)
865         CONTINUE
870      CONTINUE
       WRITE(20,*) '/ ;'
       WRITE(20,*) 'PARAMETER DIST2(J,K) DISTANCE FROM BEACH TO OBJ /'
         DO 880, P=1,ANUM2
             DO 875, Q=1,ANUM3
               WRITE(20,*) LAND(P), '.',OBJ(Q), DIST2(P,Q)
875          CONTINUE
880      CONTINUE
       WRITE(20,*) '/ ;'

      WRITE(20,*) 'PARAMETER SPEED1(I,J) SPEED FROM SHIP TO BEACH  /'
        DO 890, P=1,ANUM1
            DO 885, Q=1,ANUM2
                 WRITE(20,*) SHIP(P),'.',LAND(Q), SPEED1(P,Q)
885         CONTINUE
890      CONTINUE
       WRITE(20,*) '/ ;'
       WRITE(20,*) 'PARAMETER SPEED2(J,K) SPEED FROM BEACH TO OBJ /'
         DO 900, P=1,ANUM2
             DO 910, Q=1,ANUM3
               WRITE(20,*) LAND(P), '.',OBJ(Q), SPEED2(P,Q)
910          CONTINUE
900      CONTINUE
       WRITE(20,*) '/ ;'




      WRITE(20,*) 'PARAMETER CAS1(I,J) FROM SHIP TO BEACH  /'
        DO 915, P=1,ANUM1
            DO 912, Q=1,ANUM2
                 WRITE(20,*) SHIP(P),'.',LAND(Q), CAS1(P,Q)
912         CONTINUE
915      CONTINUE
```

```
      WRITE(20,*) '/ ;'
      WRITE(20,*) 'PARAMETER CAS2(J,K) FROM BEACH TO OBJ /'
        DO 920, P=1,ANUM2
            DO 919, Q=1,ANUM3
              WRITE(20,*) LAND(P), '.',OBJ(Q), CAS2(P,Q)
919         CONTINUE
920     CONTINUE
      WRITE(20,*) '/  ; '
      WRITE(20,*) 'PARAMETER MAXTIME(K) MAXIMUM TIME TO OBTAIN OBJ /'
        DO 921, Q=1,ANUM3
            WRITE(20,*) OBJ(Q),MAXTIME(Q)
921     CONTINUE
      WRITE(20,*) '/  ; '

      WRITE(20,*) 'PARAMETER COST1(I,J) ARC COST FROM I TO J'
      WRITE(20,*) 'COST2(J,K) ARC COST FROM J TO K ;'
      WRITE(20,*) 'COST1(I,J) = CAS1(I,J)*DIST1(I,J)/SPEED1(I,J) ;'
      WRITE(20,*) 'COST2(J,K) = CAS2(J,K)*DIST2(J,K)/SPEED2(J,K) ;'

      WRITE(20,*) 'VARIABLES X(I,J,K)  MARINES GOING FROM I TO K VIA J'
      WRITE(20,*) 'S(I,J,K) ELASTIC VARIABLE '
      WRITE(20,*) 'E  BINARY FLAG SIGNIFYING ELASTICITY IS REQUIRED'
      WRITE(20,*) 'T(I,J,K)  FLAG DENOTING AVENUE IJK IS USED'
      WRITE(20,*) 'Z  TOTAL COST THROUGH NETWORK ; '

      WRITE(20,*) 'POSITIVE VARIABLE X ; '
      WRITE(20,*) 'POSITIVE VARIABLE S ; '
      WRITE(20,*) 'BINARY VARIABLE E ; '
      WRITE(20,*) 'BINARY VARIABLE T ; '


      DO 922 P=1,ANUM1
          DO 923 Q=1,ANUM2
             DO 924 R=1,ANUM3
              IF((WX1(P,Q).GE.999.0).OR.(WX2(Q,R).GE.999.0)) THEN
                WRITE(20,950) SHIP(P), LAND(Q), OBJ(R)
              ENDIF
924       CONTINUE
923     CONTINUE
922   CONTINUE
950           FORMAT(' X.FX("',A,'","',A,'","',A,'") = 0 ;')

      COUNT = 0
      DO 926 R=1,ANUM3
          DO 927 Q=1,ANUM2
             OK = .FALSE.
             DO 928 P=1,ANUM1
              IF(DIST1(P,Q)/SPEED1(P,Q)+DIST2(Q,R)/SPEED2(Q,R)
     C           .LE. MAXTIME(R))THEN
                 OK = .TRUE.
              ENDIF
928          CONTINUE
             IF(OK) THEN
                 COUNT = COUNT + 1
             ELSE
                 WRITE(20,949) LAND(Q), OBJ(R)
             ENDIF
927     CONTINUE
926   CONTINUE
949           FORMAT(' Y.FX("',A,'","',A,'") = 0 ;')
```

```
       WRITE(20,*) 'LOSS'
       WRITE(20,*) 'SHIP(I)'
       WRITE(20,*) 'BEACH(J)'
       WRITE(20,*) 'OBJ1(K)'
       WRITE(20,*) 'OBJ2(J)'
       DO 1000 L=1,COUNT
         IF(L .LE. 9) THEN
            WRITE(20,1001)L
         ELSEIF (L .LE.99) THEN
            WRITE(20,1002)L
         ELSE
            WRITE(20,1003)
         END IF
1000   CONTINUE
1001   FORMAT(' RESTY',I1)
1002   FORMAT(' RESTY',I2)
1003   FORMAT(' RESTY',I3)
       WRITE(20,*) 'FLAG1'
       WRITE(20,*) 'FLAG2'
       WRITE(20,*) 'REST1'
       WRITE(20,*) 'TIMECHK1(I,J,K)'
       WRITE(20,*) 'TIMECHK2(I,J,K)'
       WRITE(20,*) 'REST2(I,J,K) ;'

       WRITE(20,*) 'LOSS .. Z =E= SUM((I,J,K), (COST1(I,J)*(X(I,J,K)+ '
       WRITE(20,*) '      S(I,J,K)))) + SUM((J,K), COST2(J,K)*Y(J,K)) ;'
       WRITE(20,*) 'SHIP(I) ..  SUM((J,K), X(I,J,K))  =L= A(I) ; '
       WRITE(20,*) 'BEACH(J) .. SUM((I,K),((1-COST1(I,J))*(X(I,J,K)+'
       WRITE(20,*) '                S(I,J,K)))) =G= B(J)  ;'
       WRITE(20,*) 'OBJ1(K) ..  SUM((J), Y(J,K)-(COST2(J,K)*Y(J,K)))'
       WRITE(20,*) '            =E= C(K) ; '
       WRITE(20,*) 'OBJ2(J) ..  SUM((K),Y(J,K)) =L= SUM((I,K), (1 -'
       WRITE(20,*) '            COST1(I,J)) * (X(I,J,K) + S(I,J,K))) ;'
       COUNT1= 0
       DO 1004 K=1,ANUM3
         DO 1005 J=1,ANUM2
            COUNT2 = 0
            IF(DIST2(J,K)/SPEED2(J,K) .LE. MAXTIME(K)) THEN
               DO 1006 I=1,ANUM1
                  IF(DIST1(I,J)/SPEED1(I,J)+DIST2(J,K)/SPEED2(J,K) .LT. .
     C               MAXTIME(K)) THEN
                     COUNT2 = COUNT2 + 1
                     TEMP(COUNT2,1) = I
                     TEMP(COUNT2,2) = J
                     TEMP(COUNT2,3) = K
                  ENDIF
1006           CONTINUE
               IF(COUNT2 .NE. 0) COUNT1 = COUNT1 + 1
               DO 1007 L = 1,COUNT2
                  P = TEMP(L,1)
                  Q = TEMP(L,2)
                  R = TEMP(L,3)
                  C = COUNT1
                  IF(COUNT2 .EQ. 1)THEN
                     IF(COUNT1 .LE. 9) THEN
                        WRITE(20,1008)C,LAND(Q),OBJ(R),SHIP(P),LAND(Q),
     C                  SHIP(P),LAND(Q),OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                     ELSEIF(COUNT1 .LE.99)THEN
                        WRITE(20,1009)C,LAND(Q),OBJ(R),SHIP(P),LAND(Q),
     C                  SHIP(P),LAND(Q),OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                     ELSE
                        WRITE(20,1010)C,LAND(Q),OBJ(R),SHIP(P),LAND(Q),
     C                  SHIP(P),LAND(Q),OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                     ENDIF
                  ELSEIF(L .EQ. 1)THEN
```

```
                        WRITE(20,1013)C,LAND(Q),OBJ(R),SHIP(P),LAND(Q),
     C                  SHIP(P),LAND(Q),OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                     ENDIF
                  ELSEIF(L .EQ. COUNT2)THEN
                        WRITE(20,1014)SHIP(P),LAND(Q),SHIP(P),LAND(Q),
     C                                OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                  ELSE
                        WRITE(20,1015)SHIP(P),LAND(Q),SHIP(P),LAND(Q),
     C                                OBJ(R),SHIP(P),LAND(Q),OBJ(R)
                  ENDIF
1007           CONTINUE
1008  FORMAT('RESTY',I1,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ;')
1009  FORMAT('RESTY',I2,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ;')
1010  FORMAT('RESTY',I3,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ;')
1011  FORMAT('RESTY',I1,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ')
1012  FORMAT('RESTY',I2,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ')
1013  FORMAT('RESTY',I3,' .. Y("',A,'","',A,'") =L= (1 - ',
     C       'COST1("',A,'","',A,'"))',/,36X,'*(X("',A,'","',A,'","',
     C       A,'")',/,36X,'+ S("',A,'","',A,'","',A,'")) ')
            ENDIF
1005     CONTINUE
1004   CONTINUE
       WRITE(20,*) 'FLAG1 .. E - SUM((I,J,K), S(I,J,K)) =L= 0 ;'
       WRITE(20,*) 'FLAG2 .. 100000*E - SUM((I,J,K),S(I,J,K)) =G= 0 ;'
       WRITE(20,*) 'REST1 .. SUM((I,J,K), X(I,J,K)) =G= SUM(I,A(I))*E ;'
       WRITE(20,*) 'TIMECHK1(I',J,K) ..  (DIST1(I,J)/SPEED1(I,J) + '
       WRITE(20,*) 'DIST2(J,K)/SPEED2(J,K))*T(I,J,K) =L= MAXTIME(K) ;'
       WRITE(20,*) 'TIMECHK2(I,J,K) .. -1*(DIST1(I,J)/SPEED1(I,J) + '
       WRITE(20,*) 'DIST2(J,K)/SPEED2(J,K)) =L= (T(I,J,K)-1)*'
       WRITE(20,*) '                             MAXTIME(K) ;'
       WRITE(20,*) 'REST2(I,J,K) .. X(I,J,K) =L= 100000*T(I,J,K);'
       WRITE(20,*) ' MODEL AMPHIB /ALL/ ; '
       WRITE(20,*) 'SOLVE AMPHIB USING MIP MINIMIZING Z ; '
       WRITE(20,*) 'DISPLAY T.L ;'
       WRITE(20,*) 'PARAMETER SUMMARY1(I,*) ;'
       WRITE(20,*) 'SUMMARY1(I,J) = SUM((K), X.L(I,J,K)) ;'
       WRITE(20,*) 'SUMMARY1(I, "COST ") = SUM((J,K), COST1(I,J)',
     C             '*X.L(I,J,K)) ;'
       WRITE(20,*) 'PARAMETER SUMMARY2(J,*) ;'
       WRITE(20,*) 'SUMMARY2(J, K) = Y.L(J,K) ;'
       WRITE(20,*) 'SUMMARY2(J, "COST ") = SUM(K,COST2(J,K)*Y.L(J,K)) ;'
       WRITE(20,*) 'PARAMETER SUMMARY3(I,*) ;'
       WRITE(20,*) 'SUMMARY3(I,J) = SUM((K), S.L(I,J,K)) ;'
       WRITE(20,*) 'SUMMARY3(I, "COST ") = SUM((J,K), COST1(I,J)',
     C             '*S.L(I,J,K)) ;'
       WRITE(20,*) 'DISPLAY SUMMARY1 ;'
       WRITE(20,*) 'DISPLAY SUMMARY2 ;'
       WRITE(20,*) 'DISPLAY SUMMARY3 ;'
*999   PRINT *, 'HAVE A NICE DAY'
      STOP
      END
```

APPENDIX B. GAMS FILE "AMPH GAMS A1"

```
$TITLE MARINE AMPHIB LANDING MODEL
$STITLE THESIS BY CATHY JOHNSON
$OFFSYMXREF OFFSYMLIST OFFUPPER
 OPTIONS LIMCOL = 0.0, LIMROW = 0.0, SOLPRINT = off;
 OPTIONS OPTCR = 0.0001, ITERLIM = 10000 ;
 SETS
    I  SHIPS OFF SHORE /
 IWOJMA,
 PELILU,
 TARAWA
  /
   J  LANDING BEACHES /
 RED   ,
 WHITE ,
 BLUE
 /
   K  OBJECTIVES /
 COMM  ,
 EMBASY,
 AIRFLD
 / ;
 PARAMETERS
 A(I)  NUMBER OF MARINES ON SHIP I  /
 IWOJMA        325
 PELILU        280
 TARAWA        225
 /
 B(J) NUMBER OF MARINES NEEDED AT BEACHHEAD J /
 RED           300
 WHITE         325
 BLUE          250
 /
 C(K) # OF MARINES REQUIRED TO COMPLETE MISSION /
 COMM          350
 EMBASY        225
 AIRFLD        275
 / ;
 PARAMETER DIST1(I,J) DISTANCE FROM SHIP TO BEACH  /
 IWOJMA.RED     1.80000019
 IWOJMA.WHITE   2.39999962
 IWOJMA.BLUE    5.60000038
 PELILU.RED     4.30000019
 PELILU.WHITE   2.00000000
 PELILU.BLUE    5.50000000
 TARAWA.RED     4.50000000
 TARAWA.WHITE   3.00000000
 TARAWA.BLUE    2.00000000
 / ;
 PARAMETER DIST2(J,K) DISTANCE FROM BEACH TO OBJ /
 RED   .COMM    4.00000000
 RED   .EMBASY  4.50000000
 RED   .AIRFLD  5.00000000
 WHITE .COMM    3.19999981
 WHITE .EMBASY  3.00000000
 WHITE .AIRFLD  3.50000000
 BLUE  .COMM    6.00000000
 BLUE  .EMBASY  5.19999981
 BLUE  .AIRFLD  4.10000038
 / ;
```

```
PARAMETER SPEED1(I,J) SPEED FROM SHIP TO BEACH  /
IWOJMA.RED      10.0000000
IWOJMA.WHITE    10.0000000
IWOJMA.BLUE     10.0000000
PELILU.RED      14.0000000
PELILU.WHITE    14.0000000
PELILU.BLUE     14.0000000
TARAWA.RED      12.0000000
TARAWA.WHITE    12.0000000
TARAWA.BLUE     12.0000000
/ ;
PARAMETER SPEED2(J,K) SPEED FROM BEACH TO OBJ /
RED   .COMM     5.00000000
RED   .EMBASY   5.00000000
RED   .AIRFLD   5.00000000
WHITE .COMM     4.00000000
WHITE .EMBASY   4.00000000
WHITE .AIRFLD   4.00000000
BLUE  .COMM     5.00000000
BLUE  .EMBASY   5.00000000
BLUE  .AIRFLD   5.00000000
/ ;
PARAMETER CAS1(I,J) FROM SHIP TO BEACH  /
IWOJMA.RED     0.420000000E-01
IWOJMA.WHITE   0.430000000E-01
IWOJMA.BLUE    0.430000000E-01
PELILU.RED     0.430000000E-01
PELILU.WHITE   0.420000000E-01
PELILU.BLUE    0.430000000E-01
TARAWA.RED     0.440000000E-01
TARAWA.WHITE   0.450000000E-01
TARAWA.BLUE    0.450000000E-01
/ ;
PARAMETER CAS2(J,K) FROM BEACH TO OBJ /
RED   .COMM    0.320000000E-01
RED   .EMBASY  0.330000000E-01
RED   .AIRFLD  0.340000000E-01
WHITE .COMM    0.340000000E-01
WHITE .EMBASY  0.340000000E-01
WHITE .AIRFLD  0.240000000E-01
BLUE  .COMM    0.220000000E-01
BLUE  .EMBASY  0.320000000E-01
BLUE  .AIRFLD  0.330000000E-01
/  ;
PARAMETER MAXTIME(K) MAXIMUM TIME TO OBTAIN OBJ /
COMM      6.0000000
EMBASY    8.0000000
AIRFLD    9.0000000
/  ;
PARAMETER COST1(I,J) ARC COST FROM I TO J
COST2(J,K) ARC COST FROM J TO K ;
COST1(I,J) = CAS1(I,J)*DIST1(I,J)/SPEED1(I,J) ;
COST2(J,K) = CAS2(J,K)*DIST2(J,K)/SPEED2(J,K) ;
VARIABLES X(I,J,K)  MARINES GOING FROM I TO K VIA J
Y(J,K) MARINES GOING FROM J TO K
S(I,J,K) ELASTIC VARIABLE
E  BINARY FLAG SIGNIFYING ELASTICITY IS REQUIRED
T(I,J,K)  BINARY VARIABLE DENOTING AVENUE IJK IS WITHIN MAXIMUM TIME
Z  TOTAL COST THROUGH NETWORK ;
POSITIVE VARIABLE X ;
POSITIVE VARIABLE Y ;
POSITIVE VARIABLE S ;
BINARY VARIABLE E ;
BINARY VARIABLE T ;
```

```
EQUATIONS
LOSS
SHIP(I)
BEACH(J)
OBJ1(K)
OBJ2(J)
RESTY1
FLAG1
FLAG2
REST1
TIMECHK1(I,J,K)
TIMECHK2(I,J,K)
REST2(I,J,K) ;
LOSS .. Z =E= SUM((I,J,K), (COST1(I,J)*(X(I,J,K)+
      S(I,J,K)))) + SUM((J,K), COST2(J,K)*Y(J,K)) ;
SHIP(I) ..  SUM((J,K), X(I,J,K))  =L= A(I) ;
BEACH(J) .. SUM((I,K),((1-COST1(I,J))*(X(I,J,K)+
                S(I,J,K)))) =G= B(J)  ;
OBJ1(K) ..  SUM((J), Y(J,K)-(COST2(J,K)*Y(J,K)))
            =E= C(K) ;
OBJ2(J) ..  SUM((K),Y(J,K)) =L= SUM((I,K), (1 -
            COST1(I,J)) * (X(I,J,K) + S(I,J,K))) ;
RESTY1 .. Y("RED    ","COMM   ") =L= (1 - COST1("IWOJMA","RED    "))
                                     *(X("IWOJMA","RED    ","COMM   ")
                                     + S("IWOJMA","RED    ","COMM   "))
FLAG1 .. E - SUM((I,J,K), S(I,J,K)) =L= 0 ;
FLAG2 .. 100000*E - SUM((I,J,K),S(I,J,K)) =G= 0 ;
REST1 .. SUM((I,J,K), X(I,J,K)) =G= SUM(I,A(I))*E ;
TIMECHK1(I,J,K) ..  (DIST1(I,J)/SPEED1(I,J) +
DIST2(J,K)/SPEED2(J,K))*T(I,J,K) =L= MAXTIME(K) ;
TIMECHK2(I,J,K) .. -1*(DIST1(I,J)/SPEED1(I,J) +
DIST2(J,K)/SPEED2(J,K)) =L= (T(I,J,K)-1)*
                              MAXTIME(K) ;
REST2(I,J,K) .. X(I,J,K) =L= 100000*T(I,J,K);
 MODEL AMPHIB /ALL/ ;
SOLVE AMPHIB USING MIP MINIMIZING Z ;
DISPLAY "THE TABLE BELOW (VARIABLE T.L) DISPLAYS THE BINARY VARIABLE",
        "(0/1) TO INDICATE IF THE AVENUE (IJK) IS WITHIN THE TIME ",
        "CONTRAINT SET BY THE BATTLEFIELD COMMANDER" ;
DISPLAY T.L ;
PARAMETER SUMMARY1(I,*) ;
SUMMARY1(I,J) = SUM((K), X.L(I,J,K)) ;
SUMMARY1(I, "#CASUALTY") = SUM((J,K), COST1(I,J)*X.L(I,J,K)) ;
PARAMETER SUMMARY2(J,*);
SUMMARY2(J, K) = Y.L(J,K) ;
SUMMARY2(J, "#CASUALTY") = SUM(K,COST2(J,K)*Y.L(J,K)) ;
PARAMETER SUMMARY3(I,*);
SUMMARY3(I,J) = SUM((K), S.L(I,J,K)) ;
SUMMARY3(I, "#CASUALTY") = SUM((J,K), COST1(I,J)*S.L(I,J,K)) ;
DISPLAY "THE TABLE BELOW (PARAMETER SUMMARY1) DISPLAYS THE OPTIMAL",
        "NUMBER OF MARINES TO SEND FROM SHIP I TO BEACHHEAD J";
DISPLAY SUMMARY1 ;
DISPLAY "THE TABLE BELOW (PARAMETER SUMMARY2) DISPLAYS THE OPTIMAL",
        "NUMBER OF MARINES TO SEND FROM BEACHHEAD J TO OBJECTIVE K";
DISPLAY SUMMARY2 ;
DISPLAY "THE TABLE BELOW (PARAMETER SUMMARY3) DISPLAYS THE ADDITIONAL",
        "NUMBER OF MARINES NEEDED FROM SHIP I IN ORDER TO SUCCESSFULLY",
        "COMPLETE THE MISSION";
DISPLAY SUMMARY3 ;
```

APPENDIX C. DATABASE MANAGEMENT FORM

QUESTION	INPUT TYPE	INPUT					
How many ships?	Integer value						
Names of ships?	6 char. or less						
How many beachheads?	Integer value						
Names of beachheads?	6 char. or less						
How many objectives?	Integer values						
Names of objectives?	6 char. or less						
# of Marines on ship (*i*)	Integer value						
# of Marines needed on beachhead (*j*)	Integer value						
# of Marines needed at objective (*k*)	Integer value						
Best weather estimate?	Fair, med, bad	Ship 1 → *j*					
		Ship 2 → *j*					
		Ship 3 → *j*					
		Ship 4 → *j*					
		Bchd 1 → *k*					
		Bchd 2 → *k*					
		Bchd 3 → *k*					
		Bchd 4 → *k*					
Best resistance estimate?	Low, med, high	Ship 1 → *j*					
		Ship 2 → *j*					
		Ship 3 → *j*					
		Ship 4 → *j*					
		Bchd 1 → *k*					
		Bchd 2 → *k*					
		Bchd 3 → *k*					
		Bchd 4 → *k*					

QUESTION	INPUT TYPE	INPUT					
Distance information?	real value	Ship 1 → *j*					
		Ship 2 → *j*					
		Ship 3 → *j*					
		Ship 4 → *j*					
		Bchd 1 → *k*					
		Bchd 2 → *k*					
		Bchd 3 → *k*					
		Bchd 4 → *k*					
Speed information?	real value	Ship 1 → *j*					
		Ship 2 → *j*					
		Ship 3 → *j*					
		Ship 4 → *j*					
		Bchd 1 → *k*					
		Bchd 2 → *k*					
		Bchd 3 → *k*					
		Bchd 4 → *k*					
Best terrain estimate?	Good, med, dif	Ship 1 → *j*					
		Ship 2 → *j*					
		Ship 4 → *j*					
		Ship 4 → *j*					
		Bchd 1 → *k*					
		Bchd 2 → *k*					
		Bchd 3 → *k*					
		Bchd 4 → *k*					

APPENDIX D. ECHO OF INPUT FROM FORTRAN PROGRAM

```
THE INFORMATION THAT YOU WILL NEED TO INPUT ARE THE
ESTIMATES OF WEATHER, RESISTANCE, TERRAIN, DISTANCE
AND THE SPEED OF YOUR MOVEMENTS.  REFER TO THESIS FOR
DETAILED INSTRUCTIONS (TOPIC "MODEL INPUT").
HOW MANY SHIPS ARE THERE?
?
2

INPUT THE NAMES OF THE SHIPS ON WHICH THE MARINES WILL
BE DEPARTING FROM - ABBREVIATE - 6 CHARACTERS, NO SPACES
iwojma
tarawa

HOW MANY BEACH LANDING ZONES ARE THERE?
?
2

INPUT THE LANDING ZONES OR BEACHHEADS - 6 CHARACTERS NO SPACES
red
white

HOW MANY OBJECTIVES ARE THERE?
?
3

INPUT THE OBJECTIVE NAMES - 6 CHARACTERS, NO SPACES
comm
embasy
airfld

INPUT THE NUMBER OF MARINES NEEDED ON RED
?
250

INPUT THE TIME IN HOURS FROM H-HOUR TO ACHIEVE OBJ EMBASY
?
25

INPUT THE BEST WEATHER ESTIMATE (FAIR,MED,BAD) FROM
EACH SHIP TO EACH LANDING ZONE. INPUT "NONE" IF THE
AVENUE WILL NOT BE CONSIDERED
FROM IWOJMA TO RED
fair
FROM IWOJMA TO WHITE
med

INPUT THE BEST ESTIMATE FOR RESISTANCE (LOW,MED,HIGH)
FROM SHIP TO EACH LANDING ZONE.
FROM IWOJMA TO RED
low
FROM IWOJMA TO WHITE
low
FROM WHITE  TO AIRFLD
low
```

```
INPUT THE DISTANCES FROM EACH SHIP TO EACH LANDING
ZONE. DISTANCES SHOULD BE IN MILES WITH ACCURACY AT MOST
TO TWO DIGITS.  EXAMPLE - 2.10
FROM IWOJMA TO RED
?
1.8
FROM TARAWA TO RED
?
3.5

INPUT THE BEST TERRAIN (OR SEA STATE) ESTIMATE FROM
EACH SHIP TO EACH LANDING ZONE  (GOOD,MED,DIF).
FROM IWOJMA TO RED
good
FROM IWOJMA TO WHITE
good
FROM TARAWA TO WHITE
med
FROM TARAWA TO BLUE
med

INPUT THE BEST TERRAIN (OR SEA STATE) ESTIMATE
(GOOD,MED,DIF) FROM EACH LANDING ZONE TO EACH OBJ.
FROM RED    TO COMM
good
FROM BLUE   TO COMM
goo
INCORRECT ENTRY,TRY AGAIN-GOOD,MED,DIF
good
FROM BLUE   TO EMBASY
dif
FROM BLUE   TO AIRFLD
med
END RECORDING OF TERMINAL SESSION
```

APPENDIX E. OUTPUT EXAMPLE OF OPTIMAL NUMBER OF MARINES WITH NO EXTRA MARINES REQUIRED

```
----    246 THE TABLE BELOW (VARIABLE T.L) DISPLAYS THE BINARY VARIABLE
            (0/1) TO INDICATE IF THE AVENUE (IJK) IS WITHIN THE TIME
            CONTRAINT SET BY THE BATTLEFIELD COMMANDER


----    249 VARIABLE  T.L

                       COMM        EMBASY       AIRFLD

IWOJMA.RED            1.000         1.000        1.000
IWOJMA.WHITE          1.000
IWOJMA.BLUE           1.000
PELILU.RED            1.000         1.000        1.000
PELILU.WHITE          1.000         1.000        1.000
TARAWA.RED                          1.000


----    259 THE TABLE BELOW (PARAMETER SUMMARY1) DISPLAYS THE OPTIMAL
            NUMBER OF MARINES TO SEND FROM SHIP I TO BEACHHEAD J


----    261 PARAMETER SUMMARY1

                  RED        WHITE         BLUE   #CASUALTY

IWOJMA        129.402                   220.598      11.847
PELILU        107.162      212.838                    7.202
TARAWA         39.286                                 1.684


----    262 THE TABLE BELOW (PARAMETER SUMMARY2) DISPLAYS THE OPTIMAL
            NUMBER OF MARINES TO SEND FROM BEACHHEAD J TO OBJECTIVE K


----    264 PARAMETER SUMMARY2

                COMM       EMBASY       AIRFLD   #CASUALTY

RED                       205.433       63.121       6.341
WHITE        160.115                    49.885      12.212


----    265 THE TABLE BELOW (PARAMETER SUMMARY3) DISPLAYS THE ADDITIONAL
            NUMBER OF MARINES NEEDED FROM SHIP I IN ORDER TO SUCCESSFULLY
            COMPLETE THE MISSION


----    268 PARAMETER SUMMARY3

                           ( ALL  ZERO )
```

APPENDIX F. OUTPUT EXAMPLE OF EXTRA MARINES REQUIRED AND AVENUES OF APPROACH WHICH DO NOT MEET THE TIME CONSTRAINT

```
----    246 THE TABLE BELOW (VARIABLE T.L) DISPLAYS THE BINARY VARIABLE
            (0/1) TO INDICATE IF THE AVENUE (IJK) IS WITHIN THE TIME
            CONTRAINT SET BY THE BATTLEFIELD COMMANDER


----    249 VARIABLE  T.L

                    COMM      EMBASY      AIRFLD

IWOJMA.RED         1.000       1.000       1.000
IWOJMA.WHITE       1.000
IWOJMA.BLUE        1.000
PELILU.RED         1.000       1.000       1.000
PELILU.WHITE       1.000       1.000       1.000
TARAWA.RED                     1.000


----    259 THE TABLE BELOW (PARAMETER SUMMARY1) DISPLAYS THE OPTIMAL
            NUMBER OF MARINES TO SEND FROM SHIP I TO BEACHHEAD J


----    261 PARAMETER SUMMARY1

               RED       WHITE   #CASUALTY

IWOJMA     350.000                   3.377
PELILU                 320.000       4.267
TARAWA     390.000                  16.714


----    262 THE TABLE BELOW (PARAMETER SUMMARY2) DISPLAYS THE OPTIMAL
            NUMBER OF MARINES TO SEND FROM BEACHHEAD J TO OBJECTIVE K


----    264 PARAMETER SUMMARY2

             COMM      EMBASY      AIRFLD   #CASUALTY

RED                   410.865     314.529      15.394
WHITE     373.601                              23.601


----    265 THE TABLE BELOW (PARAMETER SUMMARY3) DISPLAYS THE ADDITIONAL
            NUMBER OF MARINES NEEDED FROM SHIP I IN ORDER TO SUCCESSFULLY
            COMPLETE THE MISSION


----    268 PARAMETER SUMMARY3

               RED       WHITE        BLUE   #CASUALTY

IWOJMA       5.539                               0.053
PELILU                  58.650     324.117      14.899
```

APPENDIX G. OUTPUT EXAMPLE OF EXTRA MARINES REQUIRED AND WITH ALL AVENUES OF APPROACH VALID

```
 ----    249 VARIABLE  T.L

                     COMM      EMBASY      AIRFLD

 IWOJMA.RED         1.000       1.000       1.000
 IWOJMA.WHITE       1.000       1.000       1.000
 IWOJMA.BLUE        1.000       1.000       1.000
 PELILU.RED         1.000       1.000       1.000
 PELILU.WHITE       1.000       1.000       1.000
 PELILU.BLUE        1.000       1.000       1.000
 TARAWA.RED         1.000       1.000       1.000
 TARAWA.WHITE       1.000       1.000       1.000
 TARAWA.BLUE        1.000       1.000       1.000


 ----    259 THE TABLE BELOW (PARAMETER SUMMARY1) DISPLAYS THE OPTIMAL
             NUMBER OF MARINES TO SEND FROM SHIP I TO BEACHHEAD J


 ----    261 PARAMETER SUMMARY1

                 RED       WHITE         BLUE   #CASUALTY

 IWOJMA     251.822                    98.178       7.147
 PELILU                 93.600        226.400      11.109
 TARAWA     390.000                                16.714


 ----    262 THE TABLE BELOW (PARAMETER SUMMARY2) DISPLAYS THE OPTIMAL
             NUMBER OF MARINES TO SEND FROM BEACHHEAD J TO OBJECTIVE K


 ----    264 PARAMETER SUMMARY2

               COMM      EMBASY      AIRFLD  #CASUALTY

 RED                    308.149     314.529     12.678
 WHITE     373.601                              23.601


 ----    265 THE TABLE BELOW (PARAMETER SUMMARY3) DISPLAYS THE ADDITIONAL
             NUMBER OF MARINES NEEDED FROM SHIP I IN ORDER TO SUCCESSFULLY
             COMPLETE THE MISSION
1MARINE AMPHIB LANDING MODEL
 E X E C U T I N G


 ----    268 PARAMETER SUMMARY3

               WHITE   #CASUALTY

 PELILU     285.050       3.801
```

APPENDIX H. OUTPUT EXAMPLE FOR DISCUSSION OF SENSITIVITY ANALYSIS

```
----     246 THE TABLE BELOW (VARIABLE T.L) DISPLAYS THE BINARY VARIABLE
             (0/1) TO INDICATE IF THE AVENUE (IJK) IS WITHIN THE TIME
             CONTRAINT SET BY THE BATTLEFIELD COMMANDER


----     249 VARIABLE  T.L

                        COMM        EMBASY       AIRFLD

IWOJMA.RED             1.000        1.000        1.000
IWOJMA.WHITE           1.000        1.000        1.000
IWOJMA.BLUE            1.000        1.000        1.000
PELILU.RED             1.000        1.000        1.000
PELILU.WHITE           1.000        1.000        1.000
PELILU.BLUE            1.000        1.000        1.000
TARAWA.RED             1.000        1.000        1.000
TARAWA.WHITE           1.000        1.000        1.000
TARAWA.BLUE            1.000        1.000        1.000


----     259 THE TABLE BELOW (PARAMETER SUMMARY1) DISPLAYS THE OPTIMAL
             NUMBER OF MARINES TO SEND FROM SHIP I TO BEACHHEAD J


----     261 PARAMETER SUMMARY1

                   RED         WHITE          BLUE    #CASUALTY

IWOJMA          19.448                     330.552       20.825
PELILU                       320.000                      5.973
TARAWA         390.000                                   23.400


----     262 THE TABLE BELOW (PARAMETER SUMMARY2) DISPLAYS THE OPTIMAL
             NUMBER OF MARINES TO SEND FROM BEACHHEAD J TO OBJECTIVE K


----     264 PARAMETER SUMMARY2

                COMM        EMBASY        AIRFLD   #CASUALTY

RED                        310.630       316.632      17.262
WHITE        381.394                                  31.394


----     265 THE TABLE BELOW (PARAMETER SUMMARY3) DISPLAYS THE ADDITIONAL
             NUMBER OF MARINES NEEDED FROM SHIP I IN ORDER TO SUCCESSFULLY
             COMPLETE THE MISSION


----     268 PARAMETER SUMMARY3

                   RED         WHITE   #CASUALTY

IWOJMA         244.925                     3.438
PELILU                        68.649       1.281
```

REFERENCES

1. Moore, R. S., "Finding the GAPS: Intelligence and MAGTF Warfare," *Marine Corps Gazette*, March 1991, p. 59.

2. Griffith, S. B., *Sun Tzu: The Art of War*, Oxoford University Press, New York and Oxford, 1963.

3. Moore, R. S., "Recon-Pull: A Marriage of 2s and 3s," *Marine Corps Gazette*, August 1980, p. 71-75.

4. U.S. Marine Corps Field Manual FM 34-130, Intelligence Preparation of the Battlefield, 23 May 1991.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Cameron Station
Alexandria, VA 22304-6145 2

2. Library, Code 52
Naval Postgraduate School
Monterey, CA 93943-5002 2

3. Deputy Undersecretary of the Army
for Operations Research
Room 2E261, The Pentagon
Washington, DC 20310 1

4. Director
USMC Development and Education Center
Quantico, VA 22134 1

5. Marine Corps Operations Analysis Group
Center for Naval Analyses
4401 Ford Avenue
P.O. Box 11280
Alexandria, VA 22302-0268 1

6. Professor Samuel H. Parry, Code OR/Py
Department of Operations Research
Naval Postgraduate School
Monterey, CA 93943 1

7. Professor William Walsh, Code OR/Wa
Department of Operations Research
Naval Postgraduate School
Monterey, CA 93943 1

8. Operations Analysis Programs, Code 30
Department of Operations Research
Naval Postgraduate School
Monterey, CA 93943 1

9. Commandant of the Marine Corps
Code TE 06
Headquarters, U.S. Marine Corps
Washington, DC 20380-0001 1

Thesis
J5733 Johnson
c.1 Mathematical model of
a Marine Corps landing
based upon intelligence
estimates.

Thesis
J5733 Johnson
c.1 Mathematical model of
a Marine Corps landing
based upon intelligence
estimates.

DUDLEY KNOX LIBRARY
3 2768 00036348 5